WORLD FORUM ON ACADEMIC INNOVATION AND DIGITAL LITERACY

WEB SAYT ARXITEKTURASINI LOYIHALASHDA MA’LUMOTLAR BAZASI VA SERVER TEXNOLOGIYALARINING INTEGRATSIYASI

Arslonov Qodir Ziyodullayevich

Osiyo Xalqaro universiteti "Umumtexnik fanlar" kafedrasi o'qituvchisi

Komilov Mirshod Tangriberganovich

“Ta’limda axborot texnologiyalari” yo‘nalishi magistranti

Osiyo xalqaro universiteti

Annotatsiya: Ushbu maqolada web sayt arxitekturasini loyihalashda ma’lumotlar bazasi va server texnologiyalarining integratsiyasi ilmiy-amaliy jihatdan tahlil qilinadi. Zamonaviy web saytlar foydalanuvchi interfeysi, server mantiqi, ma’lumotlar bazasi, API, xavfsizlik va bulut infratuzilmasi o‘rtasidagi uyg‘unlik asosida ishlaydi. Maqolada arxitektura qatlamlari, SQL va NoSQL bazalar, backend texnologiyalari, kesh, autentifikatsiya, yuklama taqsimoti, xavfsiz ulanish va kengayish imkoniyatlari ko‘rib chiqiladi. Tadqiqot natijasida ma’lumotlar bazasi va server texnologiyalarining to‘g‘ri integratsiyasi web saytning tezligi, ishonchliligi, xavfsizligi va uzoq muddatli rivojlanishini ta’minlovchi asosiy omil ekanligi asoslanadi.

Kalit so‘zlar: *web arxitektura, ma’lumotlar bazasi, server texnologiyalari, backend, API, SQL, NoSQL, kesh, xavfsizlik, bulut infratuzilmasi, integratsiya.*

Аннотация: В статье анализируется интеграция баз данных и серверных технологий при проектировании архитектуры веб-сайта. Современные веб-сайты работают на основе согласованного взаимодействия пользовательского интерфейса, серверной логики, баз данных, API, безопасности и облачной

инфраструктуры. Обосновывается, что правильная интеграция является основой скорости, надежности, безопасности и масштабируемости веб-сайта.

Ключевые слова: *веб-архитектура, база данных, серверные технологии, backend, API, SQL, NoSQL, кэширование, безопасность, облачная инфраструктура.*

Abstract: The article analyzes the integration of databases and server technologies in website architecture design. Modern websites operate through the coordinated interaction of the user interface, server logic, databases, APIs, security and cloud infrastructure. The paper argues that proper integration is a key factor for website speed, reliability, security and scalability.

Keywords: *web architecture, database, server technologies, backend, API, SQL, NoSQL, caching, security, cloud infrastructure, integration.*

Web sayt yaratish jarayonida arxitektura masalasi ko‘pincha dizayn yoki sahifa tuzilishidan keyin ko‘riladigan texnik bosqich sifatida qabul qilinadi. Aslida esa web saytning barqaror ishlashi, tezligi, xavfsizligi va kelajakda kengayishi aynan arxitektura loyihasiga bog‘liq. Ayniqsa, ma’lumotlar bazasi va server texnologiyalari o‘rtasidagi integratsiya noto‘g‘ri qurilsa, sayt tashqi ko‘rinishidan yaxshi bo‘lsa ham, amaliy foydalanishda sekin, zaif yoki ishonchsiz bo‘lib qoladi.

Zamonaviy web sayt foydalanuvchi brauzerida ochiladigan sahifadan iborat emas. Uning ortida server, dasturiy mantiq, ma’lumotlar bazasi, API, autentifikatsiya moduli, fayl saqlash xizmati, kesh, monitoring va xavfsizlik mexanizmlari mavjud. Bular bir-biri bilan to‘g‘ri bog‘langanida web sayt axborot tizimi sifatida samarali ishlaydi [1].

Web arxitektura odatda bir nechta qatlamdan iborat bo‘ladi. Birinchi qatlam — foydalanuvchi interfeysi, ya’ni frontend. Ikkinchi qatlam — server tomoni, ya’ni backend. Uchinchi qatlam — ma’lumotlar bazasi. Murakkab tizimlarda bularga kesh,

fayl saqlash, xabarlar navbati, qidiruv indeksi va tahliliy modul ham qo‘shiladi. Bu qatlamlar o‘rtasidagi muloqot aniq protokollar va interfeyslar asosida quriladi [2].

Server texnologiyalari foydalanuvchidan kelgan so‘rovlarni qabul qiladi va qayta ishlaydi. Masalan, foydalanuvchi saytga kirganda, login parolini yuborganda, ma’lumot qidirganda yoki fayl yuklaganda server bu amallarni tekshiradi, biznes-mantiqqa muvofiq bajaradi va ma’lumotlar bazasiga murojaat qiladi. Agar server mantiqi tartibsiz yozilsa, ma’lumotlar bazasiga ortiqcha so‘rovlar yuboriladi, javob vaqti cho‘ziladi va tizim yuklamaga bardosh bera olmaydi [3].

Ma’lumotlar bazasi web saytning axborot markazidir. U foydalanuvchilar, rollar, sahifalar, mahsulotlar, test natijalari, buyurtmalar, xabarlar va boshqa yozuvlarni saqlaydi. PostgreSQL kabi tizimlar tranzaksiya, indeks, cheklov, ko‘rinish va murakkab so‘rovlar bilan ishlash imkoniyatiga ega [4].

NoSQL bazalar esa katta hajmdagi, yarim tuzilmali yoki tez o‘zgaradigan ma’lumotlarda samarali bo‘lishi mumkin. Masalan, foydalanuvchi faoliyati loglari, chat xabarlari, hujjat ko‘rinishidagi ma’lumotlar yoki moslashuvchan obyektlar uchun MongoDB kabi hujjatli bazalar qulay. Biroq NoSQLni tanlash relyatsion bazalardan voz kechish degani emas. Aksincha, ayrim tizimlarda SQL va NoSQL birgalikda ishlatiladi. Muhimi, ma’lumotlarning tabiati va foydalanish stsenariysi to‘g‘ri baholanishidir [5].

Server va ma’lumotlar bazasi integratsiyasida API muhim vosita hisoblanadi. API frontend va backend o‘rtasida ma’lumot almashish qoidalarini belgilaydi. REST yondashuvida resurslar URL orqali ifodalanadi, HTTP metodlari yordamida ma’lumot olinadi, yaratiladi, o‘zgartiriladi yoki o‘chiriladi. GraphQL esa foydalanuvchiga kerakli maydonlarni aniq so‘rash imkonini beradi. Har ikki yondashuvda ham autentifikatsiya, ma’lumotni tekshirish va xatoliklarni to‘g‘ri qaytarish muhim [6].

Backend texnologiyalarini tanlashda loyiha maqsadi, jamoa tajribasi, xavfsizlik talabi, tezlik, kengayish va qo‘llab-quvvatlash imkoniyati hisobga olinadi. Node.js real

vaqtga yaqin muloqot va ko‘p sonli yengil so‘rovlar uchun qulay. Django kabi frameworklar esa tez ishlab chiqish, xavfsizlik mexanizmlari va ma’lumotlar bazasi bilan ishlash uchun tayyor ORM imkoniyatlarini beradi. Express.js minimal va moslashuvchan API yaratishga xizmat qiladi. Tanlovning ilmiy asoslangan bo‘lishi web saytning uzoq muddatli ishlashiga ta’sir qiladi [7].

Kesh texnologiyasi server va ma’lumotlar bazasi o‘rtasidagi yuklamani kamaytiradi. Ko‘p so‘raladigan ma’lumotlar har safar bazadan olinmasdan, vaqtincha xotirada saqlanishi mumkin. Bu sahifa ochilish tezligini oshiradi va server xarajatlarini kamaytiradi. Biroq kesh noto‘g‘ri sozlansa, foydalanuvchiga eskirgan ma’lumot ko‘rinishi mumkin. Shuning uchun kesh siyosati ma’lumotning yangilanish davri va ahamiyatiga qarab belgilanadi [8].

Yuklama taqsimoti ham integratsiyaning muhim qismidir. Foydalanuvchilar soni ortganda bitta server barcha so‘rovlarga javob bera olmasligi mumkin. Bunday holatda load balancer, bir nechta server nusxasi, ma’lumotlar bazasi replikatsiyasi, vertikal va gorizontal kengaytirish usullari qo‘llaniladi. Bulutli infratuzilma bu jarayonlarni moslashuvchan boshqarishga yordam beradi [9].

Xavfsizlik server va ma’lumotlar bazasi integratsiyasida alohida o‘rin tutadi. Ma’lumotlar bazasiga bevosita brauzerdan murojaat qilish mumkin emas; barcha so‘rovlar server orqali tekshirilishi kerak. SQL injection, noto‘g‘ri autentifikatsiya, zaif parol, ochiq konfiguratsiya va ruxsat nazoratining yetishmasligi web tizim uchun katta xavf tug‘diradi. Shu sababli parametrli so‘rovlar, ORM, rolga asoslangan ruxsat, token xavfsizligi, shifrlash va audit jurnali qo‘llanilishi zarur [10].

Arxitektura loyihasida monitoring va texnik xizmat ko‘rsatish ham oldindan o‘ylanishi kerak. Server yuklamasi, xatoliklar, so‘rovlar va ma’lumotlar bazasi ishlashi kuzatilmasa, muammo faqat foydalanuvchi shikoyat qilgandan keyin bilinadi. Monitoring tizimlari esa muammoni erta aniqlash, sekin so‘rovlarni topish va tizim sifatini oshirishga yordam beradi [11].

Web sayt arxitekturasining istiqboli serverless, container, microservices, edge computing va sun’iy intellekt bilan bog‘liq. Biroq har bir zamonaviy texnologiyani qo‘llash shart emas. Eng muhim tamoyil — tizim vazifasiga mos, sodda, kengaytiriladigan va xavfsiz arxitekturani tanlashdir. Dissertatsiya mavzusida ham web sayt yaratish texnologiyalarining istiqboli texnologiyani ko‘paytirishda emas, balki ularni maqsadli integratsiya qilishda namoyon bo‘ladi [12].

3-jadval. Web arxitekturada server va ma’lumotlar bazasi integratsiyasining tarkibiy qismlari

Tarkibiy qism	**Vazifasi**	**Integratsiyadagi ahamiyati**
Backend server	So‘rovlarni qabul qilish va biznes-mantiqni bajarish	Frontend va baza o‘rtasidagi asosiy bog‘lovchi qatlam
Ma’lumotlar bazasi	Axborotni saqlash, izlash va yangilash	Tizimning axborot markazi
API	Qatlamlar o‘rtasida standart aloqa yaratish	Tizimni kengaytirish va boshqa servislar bilan bog‘lashga yordam beradi
Kesh	Tez-tez ishlatiladigan ma’lumotni vaqtincha saqlash	Tezlikni oshiradi va bazaga yuklamani kamaytiradi
Xavfsizlik moduli	Ruxsat, shifrlash va tekshiruvlarni boshqarish	Ma’lumotlar maxfiyligi va tizim ishonchliligini ta’minlaydi

Samarali web arxitektura yaratish uchun ma’lumot oqimi quyidagicha rejalashtiriladi: foydalanuvchi so‘rovi → frontend validatsiya → API endpoint → server tekshiruvi → biznes-mantiq → ma’lumotlar bazasi so‘rovi → natijani qayta

ishlash → javobni foydalanuvchiga chiqarish. Ushbu ketma-ketlikda har bir bosqichning vazifasi aniq bo‘lsa, tizimni rivojlantirish va xatolarni tuzatish osonlashadi.

Masalan, ta’limga oid web saytda talaba test topshiradi. Frontend javoblarni qabul qiladi, backend ularni tekshiradi, ma’lumotlar bazasi natijani saqlaydi, server umumiy ballni hisoblaydi va foydalanuvchi kabinetida natija ko‘rsatiladi. O‘qituvchi panelida esa barcha talabalar natijasi tahliliy jadval va diagramma ko‘rinishida chiqadi. Bunday jarayon ma’lumotlar bazasi va server texnologiyalarining to‘g‘ri integratsiyalashganini ko‘rsatadi.

Foydalanilgan adabiyotlar:

1. Amazon Web Services. AWS Well-Architected Framework. AWS Documentation, 2024.
2. Google Cloud. Well-Architected Framework: Architecture Framework. Google Cloud Documentation, 2025-2026.
3. IETF. RFC 9110: HTTP Semantics. Internet Engineering Task Force, 2022.
4. PostgreSQL Global Development Group. PostgreSQL Documentation, Version 17/18. 2024-2026.
5. MongoDB. Schema Design Patterns: Database Manual. MongoDB Documentation, 2025.
6. Node.js Project. Node.js Learn Documentation. 2024-2026.
7. Django Software Foundation. Django Documentation and Django 5.1 Release Notes. 2024.
8. Express.js. Express Web Application Framework Documentation. 2024-2026.
9. OWASP Foundation. OWASP Top 10 Web Application Security Risks. 2021/2025 edition.
10. NIST. The NIST Cybersecurity Framework (CSF) 2.0. NIST CSWP 29, 2024.

11. Google. Web Vitals and Interaction to Next Paint. web.dev Documentation, 2024.
12. ISO/IEC. ISO/IEC 25010:2023 Systems and software engineering — Product quality model. Geneva: ISO, 2023.